SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE i RAČUNARSTVA

ZAVRŠNI RAD br. 3946

Lokalizacija objekata primjenom stereoskopske rekonstrukcije

Petra Marče

Zagreb, lipanj 2015.

Sadržaj

1. Uvod

Kako je u životinjskom svijetu izdvajanje ključnih informacija iz okoliša bilo preduvjet za preživljavanje, detekcija i prepoznavanje objekata postali su jedno od najvažnijih primjena osjetila vida zbog čega je vid visoko evoluirao. Čovjek može razlikovati više od trideset tisuća vizualnih kategorija i detektirati objekte u rasponu od nekoliko stotina milisekundi. Unatoč tome, još uvijek nije poznat način na koji naš mozak obrađuje vizualne informacije. Zbog toga je računalni vid znanstveno područje koje se danas intenzivno istražuje, a detekcija i prepoznavanje objekata su kao i u živom svijetu od posebnog značaja zbog široke palete primjena. Jedna od tih primjena je u prometu gdje se sustavi za detekciju mogu koristiti za bolje održavanje cesta ali i u vozilima pružajući informacije vozaču koje je možda previdio zbog umora ili nepažnje. Takvi sustavi imaju za cilj povećati sigurnost svih sudionika u prometu.

Jedan od pristupa lokalizaciji objekata je korištenje stereoskopske rekonstrukcije gdje se pomoću para slika koje snima stereo sustav kamera, rekonstruira 3D scena. Postupak rekonstrukcije sastoji se od pronalaska korespondentnih točaka i izračuna disparitetne mape iz koje je, uz poznavanje parametara kamera stereo sustava, moguće rekonstruirati strukturu oblaka točaka.

Ovaj rad se nastavlja na diplomski rad kolege Viktora Brauta koji je razvio računalni program za detekciju objekata primjenom stereoskopske rekonstrukcije iz oblaka točaka pri čemu je naglasak bio stavljen na lokalne metode podudaranja radi detekcije u realnom vremenu. U ovom radu naglasak će biti na globalnim metodama podudaranja koje iako su sporije daju bolju aproksimaciju scene i time potencijalno više detekcija. U okviru rada provela sam evaluaciju njegovog programa, podešenog da radi s globalnom metodom podudaranja na ispitnom skupu KITTI object te validirala rezultate. Cilj rada bio je upoznati se kako taj program radi te koliko dobro radi odnosno izmjeriti koliko je pouzdan i može li se eventualno koristiti u kombinaciji s drugim sofisticiranijim metodama detekcije.

U drugom poglavlju opisani su algoritmi i matematičke metode koje se koriste za rekonstrukciju scene i dobivanje mape dispariteta. U trećem poglavlju opisan je način rada programa za detekciju. U četvrtom poglavlju objašnjeni su pojmovi vezani za evaluaciju, opisan ispitni skup KITTI te način evaluacije. U petom poglavlju prikazani su rezultati evaluacije.

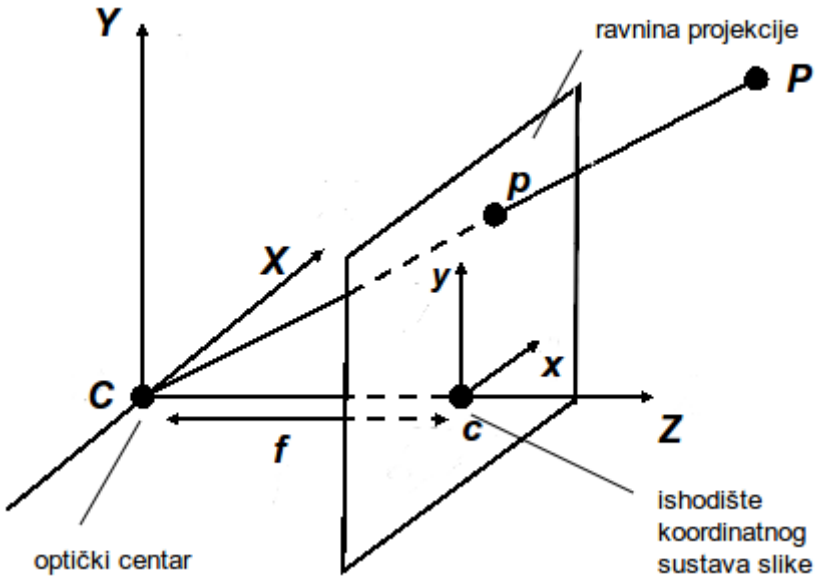
2. Stereoskopska rekonstrukcija

Koristeći kameru kao instrument koji projicira trodimenzionalne točke stvarnog svijeta u dvodimenzionalne točke slike, gubi se informacija dubine. Problem rekonstrukcije scene iz mape 2d točaka zapravo je problem određivanja dubine svake točke slike. Postavlja se pitanje, kako onda iz slikovne informacije uopće rekonstruirati scenu. Odgovor koji na to daje stereo vid je istovremeno koristiti više različito postavljenih kamera. Najjednostavniji stereo sustav je sustav od dvije međusobno razmaknute kamere . Na temelju slika koje su snimljene takvim sustavom i poznavanja međusobnog odnosa kamera moguće je dobiti informacije o 3D sceni te je rekonstruirati. Kalibracija je postupak izračuna parametara koji karakteriziraju same kamere i njihov međusobni odnos. Nakon kalibracije slijedi rektifikacija slika koja korespondentne točke u slikama postavlja na istu visinu unutar slike. Problem stereo rekonstrukcije u grubo se dijeli na problem nalaženja korespondentnih piksela u lijevoj i desnoj slici te problem reprojekcije. Problem pronalaženja podudarnih točaka može se promatrati kao minimizacijski problem pri čemu lokalne metode nastoje minimizirati više odvojenih funkcija energije koje uzimaju u obzir cijene podudaranja piksela u oknu oko promatranog piksela pri čemu je odabir veličine okna kritičan. Globalne metode ne rade s oknima već pokušavaju minimizirati globalnu funkciju energije čime se u obzir uzmiaju svi pikseli slike. Rezultat procesa određivanja podudarnih piksela je funkcija koja svakoj točki slike pridružuje disparitet odnosno horizontalnu udaljenost korespondentnog piksela u drugoj slici. Iz poznate mape dispariteta i parametara koji dobivenih kalibracijom moguće je procesom triangulacije rekonstruirati 3D scenu.

Model kamere

Kamera je uređaj koji projicira točke 3D prostora u ravninu slike. Matematički model koji opisuje to projiciranje naziva se model idealne kamere. Idealna kamera (engl. *pinehole camera*) je pojednostavljeni model realne kamere

koji se sastoji od kutije koja na jednoj svojoj strani ima rupicu kroz koju ulazi svjetlost i projicira obrnutu sliku na drugoj strani kutije. Sljedeća slika prikazuje model kamere gdje je ravnina projekcije postavljena ispred centra projekcije da bi se izbjegla obrnuta slika. Ishodište sustava kamere nalazi se u ishodištu koordinatnog sustava svijeta.



Slika 2.1. Geometrija idealne kamere

Žarišna duljina f je udaljenost ravnine projekcije od točke C koja označava projekcijski centar kamere koji se u modelu kamere postavlja u ishodište koordinatnog sustava svijeta. Os koja spaja optički centar s ravninom projekcije i na nju je okomita naziva se optička os. Točka u kojoj se optička os i ravnina projekcije sijeku predstavlja ishodište koordinatnog sustava slike.
Točka $P(X, Y, Z)$ se iz prostora perspektivnom projekcijom projicira u točku $p(x, y)$ ravnine slike. Odnos te dvije točke dan je sljedećim jednadžbama:

$$x = \alpha \frac{X}{Z} + x_0 \tag{2.1}$$

$$y = \beta \frac{Y}{Z} + y_0 \tag{2.2}$$

gdje je $\alpha = kf$, $\beta = lf$, pri čemu je f fokus, $\frac{1}{k}$ i $\frac{1}{l}$ širina i visina slike kamere, a x_0 i y_0 koordinate centra slike, ako pretpostavimo da se ishodište sustava slike nalazi u

gornjem lijevom kutu slike. Prelaskom na homogene koordinate i matrični zapis:

$$s\vec{x} = A\vec{X} \implies s\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & x_0 & y_0 \\ 0 & \beta & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.3}$$

gdje je $s = Z$, informacija koja se gubi projekcijom. Matrica A naziva se matrica kamere.

$$A = \begin{bmatrix} \alpha & 0 & x_0 & y_0 \\ 0 & \beta & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.4}$$

Vrijednosti α, β, x_0 i y_0 u matrici kamere ni na koji način ne ovise o poziciji i orijentaciji kamere u prostoru zbog čega se nazivaju intrinzičnim parametrima kamere.

Ako centar projekcije kamere sa slike 2.1. ne bi bio u ishodištu koordinatnog svijeta što je realan slučaj, potrebna nam je matrica koja će točke iz koordinatnog sustava svijeta prebaciti u sustav kamere. Takva matrica naziva se matrica ekstrinzičnih parametara $[R|t]$.

$$[R|t] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.5}$$

Matrica $[R|t]$ sastoji se od matrice rotacije R i translacijskog vektora t i ona definira položaj i orijentaciju kamere u odnosu na koordinatni sustav svijeta

Nova veza između točke svijeta i točke u slici koja je njena projekcija glasi:

$$s\vec{x} = A[R|t]\vec{X} \implies s\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & x_0 & y_0 \\ 0 & \beta & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.6}$$

Matrica $P = A[R|t]$ naziva se projekcijska matrica.

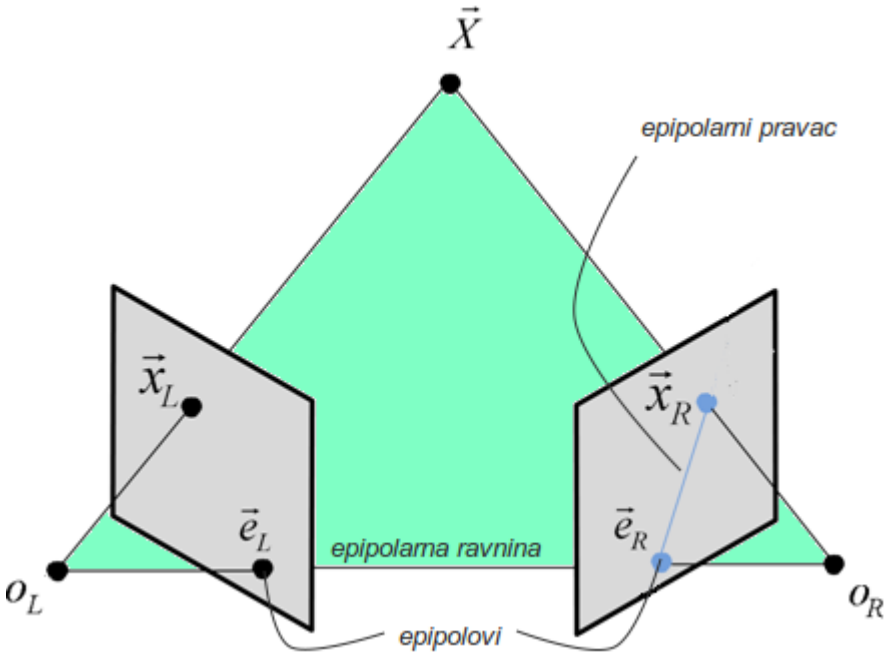
Matrica ekstrinzičnih parametara transformira točke iz 3D sustava svijeta u 3D sustav kamere, i ovisi o međusobnom položaju tih dvaju sustava, a matrica intrinzičnih parametara transformira točke iz 3D sustava kamere u 2D sustav slike.

Kalibracija stereo sustava

Postupak kojim se izračunavaju intrinzični i ekstrinzični parametri kamere naziva se kalibracija kamere. Osim do sad spomenutih linearnih intrinzičnih parametara postoje i nelinearni intrinzični parametri. Nelinearni intrinzični parametri su koeficijenti koji opisuju efekt radijalne i tangencijalne distorzije. Radijalna distorzija pojavljuje se kao posljedica ovisnosti prijeloma zrake na leći o mjestu upada zrake. Tangencijalna distorzija je uzrokovana greškom u poravnanju leće i senzora kamere koji u realnoj fizičkoj izvedbi nisu savršeno paralelni. Da bismo kalibrirali stereo sustav potrebno je pronaći matrice koje opisuju položaj jedne kamere u odnosu na drugu. Takve dvije matrice nazivaju se translacijska matrica T i rotacijska matrica R a određuju se na temelju matrica Rt obiju kamera i činjenice da kamere gledaju isti položaj kalibracijske plohe.

Epipolarna geometrija

Epipolarna geometrija opisuje geometriju scene stereo vida. Sljedeća slika prikazuje jedan stereo sustav koji snima scenu. Optički centar lijeve kamere označen je s O_L, a desne s O_R a točka koju one gledaju označena je točkom $\vec{X}$. Te tri točke razapinju epipolarnu ravninu. Projekcija točke $\vec{X}$ prostora scene na lijevu sliku je točka $\vec{x_L}$, a na sliku desne $\vec{x_R}$. Točke $\vec{e_L}$ i $\vec{e_R}$ nazivaju se epipolovima i predstavljaju sjecište pravca $O_L O_R$ te lijeve, odnosno desne ravnine projekcije. Dužina koja spaja optičke centre naziva se engl. *baseline*.



Slika 2.2 Epipolarna ravina

Dok lijeva kamera pravac $O_L\vec{X}$ vidi kao točku $\vec{x_L}$, desna kamera taj isti pravac vidi kao pravac $\vec{e_R}\vec{x_R}$. Isto tako desna kamera vidi pravac $O_R\vec{X}$ kao točku $\vec{x_R}$ dok lijeva kamera taj pravac vidi kao $\vec{e_L}\vec{x_L}$. Pravci $\vec{e_R}\vec{x_R}$ i $\vec{e_L}\vec{x_L}$ nazivaju se epipolarni pravci. Bitno je naglasiti da položaj epipolarnih pravaca ovisi o položaju točke $\vec{X}$. Ako je međusobna translacija i rotacija između kamera poznata tada epipolarna geometrija donosi sljedeća opažanja:

Ako pretpostavimo da znamo koordinate točke $\vec{x_L}$ na lijevoj slici, i zanimaju nas koordinate njoj pripadne točke $\vec{x_R}$ na desnoj slici tu točku možemo naći isključivo duž epipolarnog pravca $\vec{e_R}\vec{x_R}$. Ograničenje koje koje kaže da se točka $\vec{x_L}$ u lijevoj slici može preslikati isključivo u točku koja leži na epipolarnom pravcu naziva se epipolarno ograničenje. To ograničenje bitno smanjuje prostor pretrage korespondentne točke što omogućuje veću preciznost i bolje performanse.

Ako su poznate točke $\vec{x_L}$ i $\vec{x_R}$tada su poznati i njihovi projekcijski pravci. Ako su te dvije točke korespondentne odnosno prikazuju istu točku u prostoru tada se njihovi projekcijski pravci moraju sjeći u točki $\vec{X}$ što nam omogućuje da iz dvije korespondentne točke na slikama izračunamo točku u prostoru koju one predstavljaju. Matrice koje daju matematički opis epipolarnog ograničenja i pomoću kojih se računaju jednadžbe epipolarnih pravaca nazivaju se esencijalna i

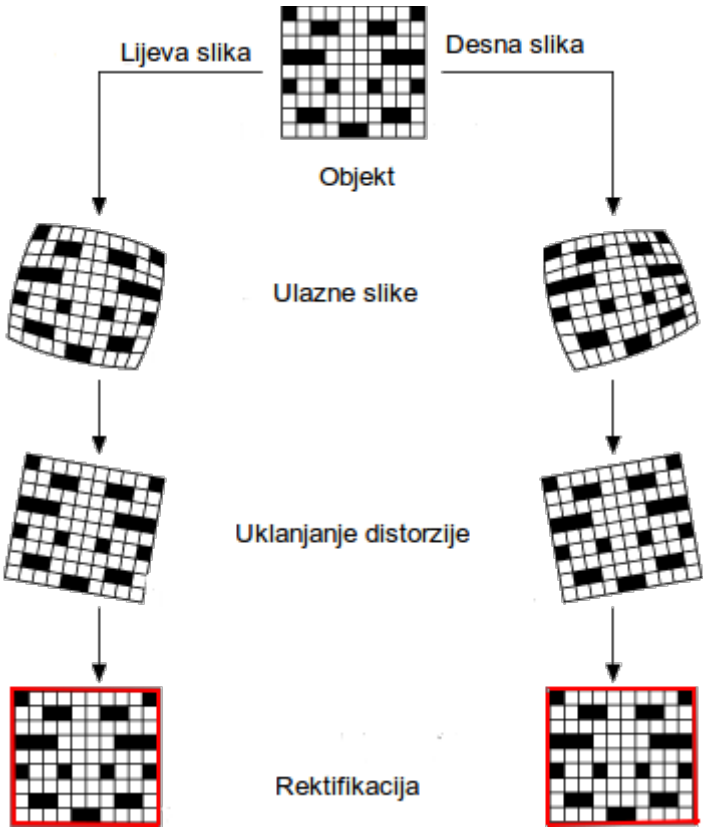
fundamentalna matrica. Esencijalna matrica sadrži informacije o translaciji i rotaciji koje vežu dvije kamere u prostoru, a fundamentalna matrica uz to sadrži i intrinzične parametre obje kamere te povezuje točke lijeve i desne slike.

Triangulacija

Triangulacija je postupak postupak pronalaženja koordinata 3d točke koordinatnog sustava svijeta na temelju točaka u lijevoj i desnoj slici koje su njena projekcija. Za idealan model kamere rješenje tog problema moguće je izvesti iz epipolarnog ograničenja koje kaže da se projekcijski pravci dviju korespondentnih točaka na slikama sigurno sijeku u točki čiju projekciju predstavljaju. Zbog činjenice da realna digitalna kamera stvara diskretiziranu sliku, preciznost samog projiciranja točke u prostoru na ravninu slike određena je rezolucijom slike. Zbog toga toga umjesto točnih projekcija dobivamo pomaknute projekcije čiji se projektivni pravci u općem slučaju ne sijeku već mimoilaze. Najčešći način izračuna točke sjecišta u takvom slučaju je uzeti polovište dužine koja predstavlja najkraću spojnicu između ta dva mimoilazeća pravca.

Rektifikacija

Rektifikacija slika je proces transformacije kojom se dobivaju slike koje bi se dobile kad bi projekcijske ravnine kamera ležale paralelno jednu uz drugu na istim visinama prije čega se uklanja efekt distorzije. Kamere se u taj položaj ne postavljaju fizički prilikom snimanja scene jer tada možda ne bi dovoljno dobro snimile područje od interesa. Nakon rektifikacije projekcijske ravnine kamera postaju paralelne s pravcem koji spaja njihove optičke centre a epipolovi se nalaze u beskonačnosti po x osi.

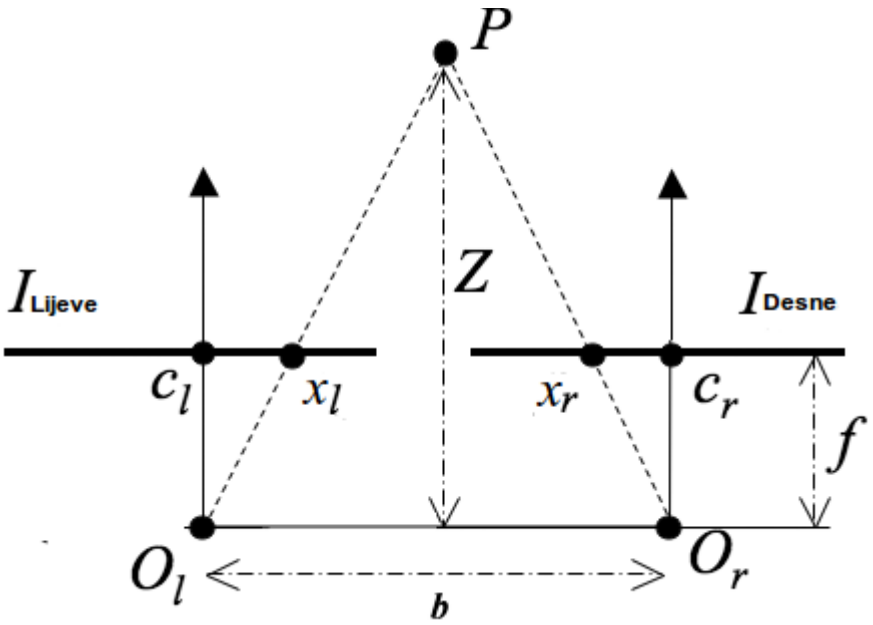


Slika 2.3 Proces rektifikacije slika

Rezultat je da su epipolarni pravci sada paralelni pravci na istoj visini unutar obje slike čime se izbjegava računanje epipolarnih pravaca i time dodatno pojednostavljuje problem nalaženja korespondentnih točaka. Sama rektifikacija se provodi uz pomoć fundamentalne matrice rotacijom kamera u traženu poziciju. Sljedeća slika prikazuje proces triangulacije nakon rektifikacije slika. Zbog sličnosti troukta vrijede relacije:

$$\frac{b}{Z} = \frac{b - x_L + x_r}{Z - f} \implies Z = \frac{f \cdot b}{x_L - x_R} = \frac{f \cdot b}{d}$$
$$\frac{X}{Z} = \frac{x_L}{f} \implies X = \frac{x_L}{f} \cdot \frac{f \cdot b}{d} = \frac{x_L \cdot b}{d} \qquad (2.7)$$
$$\frac{Y}{Z} = \frac{y_L}{f} \implies Y = \frac{y_L}{f} \cdot \frac{f \cdot b}{d} = \frac{y_L \cdot b}{d}$$

gdje je $d = x_L - x_R$ uvedena veličina i predstavlja disparitet što je razlika x koordinata korespondentnih točaka u lijevoj i desnoj slici.



Slika 2.4. Rektificirana triangulacija

Metode podudaranja

Korištenjem prethodno definiranih ograničenja, pokušava se rješiti problem podudaranja odnosno naći podudarne piksele u slikama te za njih odrediti disparitet. Rezultat tog postupka je mapa dispariteta koja svakoj točki (x, y) slike pridružuje odgovarajući disparitet $d(x, y)$. Mjera koja govori koliko se dobro dvije točke podudaraju odnosno u kojoj mjeri su korespondentne naziva se mjera sličnosti ili cijena podudaranja (*engl. Matching cost*). Osnovna mjera sličnosti je suma kvadrata udaljenosti:

$$C_{SSD}(x, d) = \sum_i [I_R(x_i + d) - I_L(x_i)]^2 \qquad (2.8)$$

gdje je x piksel na desnoj slici I_R i na lijevoj slici I_L a d disparitet. Za ovu mjeru optimalna vrijednost je nula a loše podudarnosti rezultiraju velikim cijenama. Ova metoda koristi radi svoje jednostavnosti i niske računalne cijene.

Druga, robusnija mjera kod prisutnosti šuma u slici koja raste sporije od kvadratne SSD mjere je suma apsolutnih udaljenosti koja raste linearno s pogreškom između prozora u slikama čime smanjuje utjecaj neusklađenih piksela:

$$C_{SAD}(x,d) = \sum_i |I_R(x_i + d) - I_L(x_i)| \tag{2.9}$$

Lokalne metode podudaranja procjenjuju u kojoj mjeri točka jedne slike odgovara točki druge slike zbrajajući cijenu podudaranja svih piksela u oknu oko promatrane točke. Izračun cijene podudaranja provodi za sve moguće disparitete nakon čega se traženi disparitet odabere po načelu *winner takes all* odnosno uzima se disparitet za kojeg je agregirana vrijednost cijene podudaranja minimalna.

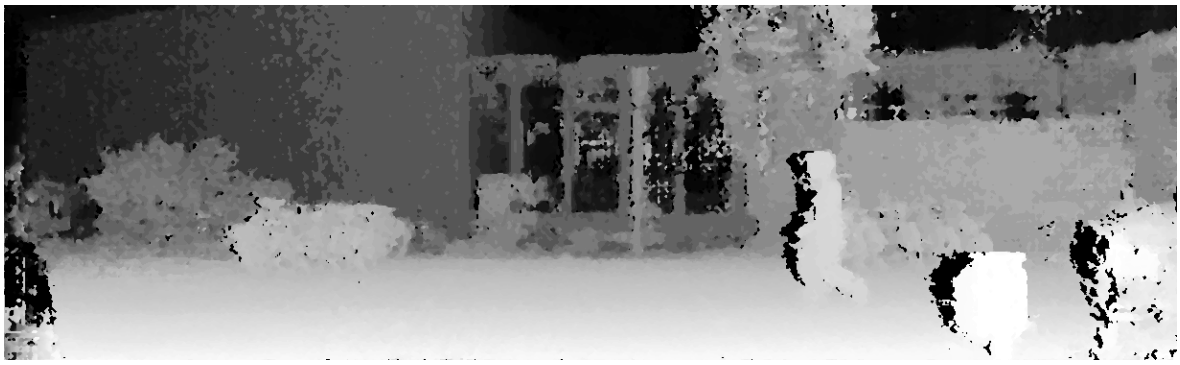
Globalne metode ne rade s oknima već minimiziraju funkciju cilja u cijeloj slici. Funkcija energije se definira kao:

$$E(d) = E_{data} + \lambda E_{smooth} \tag{2.10}$$

gdje član E_{data} predstavlja agregiranu cijenu svih piksela u slici a E_{smooth} omogućuje zaglađivanje na mjestima gdje susjedi oko promatranog piksela imaju različit disparitet. Minimizacija takve funkcije u 2D prostoru je problem NP složenosti. U okviru rada koristi se **Semi-global block matching** što je OpenCV implementacija algoritma poluglobalnog podudaranja predstavljenog u (Hirschmuller, 2005) koji uspjšno kombinira globalne i lokalne metode podudaranja.

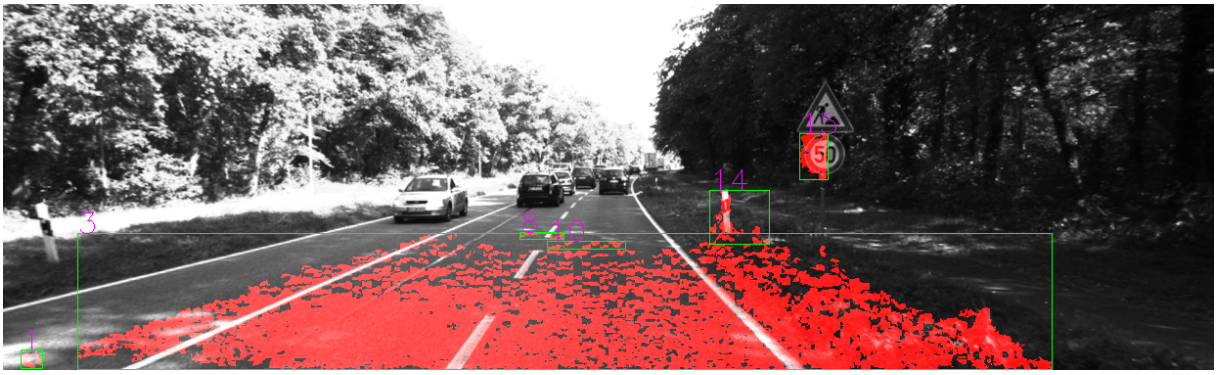
3. Program za detekciju objekata

Program koji je izgradio kolega Viktor Braut kao ulaz prima mapu dispariteta dobivenu *Semi Global Block Matching* algoritmom.



Slika 3.1. Prikaz mape dispariteta

Mapa dispariteta je matrica koja za svaki piksel lijeve slike sadrži udaljenost pripadnog piksela u desnoj slici. Koristeći prepostavku da su slike prethodno rektificirane triangulaciju točaka provodi izrazima (2.7) koristeći poznate parametre kamere i mapu dispariteta. Nakon toga poravnanva oblak točaka tako da optičke osi kamera budu paralelne s cestom i da ravnina ceste leži u xz ravnini. Kada to ne bi bio slučaj postojala bi mogućnost da se cesta detektira kao jedan veliki objekt što je prikazano slikom 3.2.



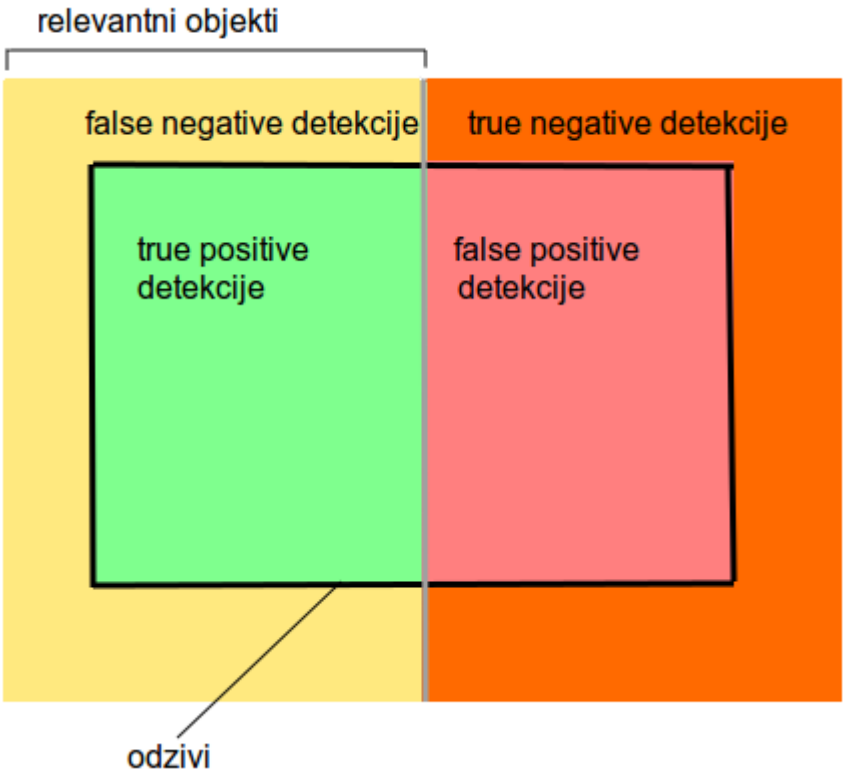
Slika 3.2. Pogrešno detektirana cesta

Da bi se to ostvarilo potrebno je prvo estimirati ravninu ceste. Cesta se estimira koristeći RANSAC metodu opisanu u (Braut, 2014). Estimacija ceste se ne vrši na temelju cijelog oblaka točaka već nekog njegovog podskupa koji najbolje

obuhvaća samu ravninu ceste.
Kako je analiza svih točaka u oblaku računski složena prostor se rešetkasto dijeli u blokove čime se dobiva prostor smanjene rezolucije. Nakon toga je za svaku interesnu točku oblaka potrebno odrediti kojem bloku u toj podjeli pripada. Izračunavaju se sferne koordinate svake točke za sustav kojemu je ishodište jednako poziciji kamere te se pridružuju odgovarajućem bloku. Pri tome je važno naglasiti da nisu sve točke u prethodno stvorenom oblaku točaka točke od interesa već samo one koje se nalaze u rasponima u kojima je smisleno tražiti objekte. Primjerice, nije smisleno tražiti objekte koji se nalaze 5 m iznad ceste jer tamo vjerojatno nema sudionika u prometu. Kolega Braut za taj raspon odredio analizirajući sekvencu slika na kojoj je provodio testiranje. Odabir preširokog raspona interesa može rezultirati većim brojem krivih detekcija, dok odabir preuskog može zanemariti neki objekt kojeg bi trebalo detektirati. Iz tog razloga sam, pri evaluaciji, kao parametar koji kontrolira broj detekcija uzela raspon područja interesa ($y_{min}, y_{max}, x_{min}, x_{max}, z_{min}$ i z_{max}).
Nadalje, nakon što se mreža popuni na prethodno opisan način, blokovi se grupiraju u objekte. Svaki blok koji sadrži više točaka od zadanog praga proglašava se zauzetim. Za svaki zauzeti blok se u traže njegovi zauzeti susjedi i te se postupak ponavlja za svaki novopronađeni zauzeti blok. Objekti koji manje od četiri bloka smatraju se šumom te se odbacuju.
Rezultati detekcije se prikazuju se na lijevoj ulaznoj slici u obliku pravokutnika koji uokviruju nađene objekte.

4. Evaluacija

Evaluacija općenito predstavlja opis, analizu i ocjenu projekata, procesa ili organizacijskih jedinica po jasno utvrđenom kriteriju. U okviru ovog rada evaluacija se provodi nad izgrađenim programom za detekciju objekata pri čemu je cilj utvrditi s kojom pouzdanošću program radi pri čemu se sve detekcije uspoređuju sa skupom unaprijed označenih okana. Takve oznake predstavljaju skup svih relevantnih detekcija (engl. *Ground truth data*).



Slika 4.1. Vrste detekcija

Sada je moguće definirati 4 vrste događaja:

- true positive (TP) – povoljan događaj, program je detektirao objekt koji se nalazi u skupu relevantnih objekata
- false positive error (FP) – program je detektirao objekt koji se ne nalazi u skupu relevantnih objekata odnosno detekcija je pogrešna
- false negative error (FN) – program nije detektirao neki od relevantnih objekata, odnosno program je učinio popust
- true negative (TN) – točno odbačena detekcija

Treba primjetiti da su svi odzivi pozitivne detekcije, koje mogu biti točne i pogrešne. Ono što nije odziv programa smatra se negativnim detekcijama. Evaluacija se provodi na način da se za svaku detekciju programa usporedbom s relevantnim skupom utvrđuje je li ona TP ili FP. Svaka oznaka okna kojoj nije pridružena detekcija programa tada je FN. Detekcija se prihvaća odnosno broji kao TP detekcija se prihvaća ako se pravokutnik koji označava detekciju(engl. *Bounding box)* preklapa s pravokutnikom oznake u određenoj mjeri. Pri tome je mjera preklapanja definirana kao kvocijent površine presjeka i površine unije ta dva pravokutnika (*engl. Intersection over union*).
Sada je moguće definirati dvije veličine kojima se opisuje pouzdanost sustava:
Odziv(engl. *recall*) :

$$R=\frac{TP}{TP+FN}$$

Preciznost(engl. *Precision*):

$$P=\frac{TP}{TP+FP}$$

Iz formula je vidljivo da je odziv veličina koja pokazuje u kojoj mjeri je sustav detektirao ono što je trebao, a preciznost veličina koja pokazuje koliko je ono što je sustav detektirao relevantno. Krivulja koja prikazuje odnos preciznost i odziva naziva se engl. *precision recall krivulja* (PR curve). Površina ispod PR krivulje je srednja preciznost.

Ispitni skup KITTI

Ispitni skup KITTI object sastoji se od 7481 slika za treniranje sustava te 7518 slika za testiranje sustava iz različitih sekvenci snimljenih u gradskoj i izvangradskoj vožnji s stereo sustavom pričvršćenim na krov vozila. Ovaj ispitni skup namijenjen je sustavima za detekciju objekata i estimaciju orijentacije objekata. Ideja ovog ispitnog skupa je da korisnici treniraju svoje programe na skupu slika za treniranje, detekcije spreme u unaprijed zadanom obliku, te pošalju na njihov evaluacijski server koji provede testiranje te korisniku dojavi rezultat. U okviru ovog rada, evaluaciju sam provela tako da sam koristila isključivo training slike i njihove oznake koje sam usporedila s izlazima programa za detekciju

implementiranog u (Brauta,2014.). Skup oznaka jedne slike skupa za treniranje dostupan je kao jedna datoteka čiji retci predstavljaju oznaku za pojedini objekt. Jedan takav redak sastoji se od 16 polja odvojenih razmacima.

Tablica 4.1. Sadržaj retka jedne oznake KITTI object skupa

Broj vrijednosti	Naziv polja	Opis
1	type	Opisuje tip označenog objekta (Car, Van, Truck, Pedestrian, Person_sitting, Cyclist, Tram, Misc or DontCare)
1	truncated	Realan broj 0-1,u kojoj mjeri objekt izlazi iz okvira slike
1	occluded	Cijeli broj 0,1,2 ili 3 u kojoj mjeri je objekt vidljiv (0-potpuno,2-uopće nije,3-nepoznato)
1	alpha	Kut promatranja objekta [-π,π]
4	tBox	Okno odnosno pravokutnik koji opisuje detekciju. Gornji lijevi i donji desni kut. (x_1, y_1, x_2, y_2)
3	dimensions	3D dimenzije objekta izražene u metrima (visina, širina, duljina)
3	location	3D lokacija objekta u koordinatama kamere izražena u metrima
1	rotation_y	Rotacija oko y osi u koordinatama kamere [-π,π]

Tablica 4.1 prikazuje koja su ta polja pri čemu broj vrijednosti označava broj stupaca koji polje obuhvaća. Vrijednosti su navedene istim redoslijedom kao i u retku datoteke oznake. Žutom bojom su u tablici naglašene vrijednosti koje sam koristila u evaluaciji.

Program za detekciju modificirala sam tako da kao igenerira po jednu tekstualnu datoteku za svaku sliku čiji su retci pravokutnici detektiranih objekta. Evaluacija se nakon toga svela na usporedbu niza korespondentnih datoteka iz izlaznog skupa programa i KITTI skupa oznaka. Pri tome se kao mjera podudaranja uzima minimalno preklapanje ili *intersection over union*. Dakle dva pravokutnika se podudaraju ako im je preklapanje veće od minimalno zadanog.

Program za evaluaciju može se pokrenuti naredbom:

```
./kitti_eval -g[putanja do mape s oznakama]  -d[putanja do mape s detekcijama] -f[putanja do liste s imenima datoteka koje treba evaluirati] -o[minimalno preklapanje u %] -n[ime razreda za evaluaciju] -m[težina evaluacije]
```

ili naredbom:

```
./kitti_eval -c[putanja do konfiguracijske datoteke]
```

Pri tome ime razreda za evaluaciju može biti Car, Cyclist ili Pedastrian, a težina evaluacije može biti 0,1 ili 2. Ideju da omogućim više razina težine evaluacije preuzela sam iz kitti evaluacijskog programa:

```
enum DIFFICULTY{EASY=0, MODERATE=1, HARD=2};
const int MIN_HEIGHT[3]             = {40, 25, 25};
const int32_t MAX_OCCLUSION[3]  = {0, 1, 2};
const double  MAX_TRUNCATION[3] = {0.15, 0.3, 0.5};
```

Program zanemaruje sve oznake i sve detekcije čija je visina pravokutnika manja od minimalne veličine za zadanu težinu evaluacije. Također ukoliko je objekt zaklonjen ili izvan slike u većoj mjeri od maksimalne propisane za zadanu težinu evaluacije, zanemaruje se. Kitti evaluacijski server implicitno za razred Car traži minimalno preklapanje od 70% a za razrede Pedestrian i Cyclist preklapanje od 50%. Moj program za evaluaciju minimalno traženo preklapanje prima kao argument komandne linije kako bi se na jednostavan način generirale krivulje ovisnosti odziva i preciznosti o minimalnom preklapanju. Program nakon što pročita parametre iz konfiguracijske datoteke ili iz komandne linije poziva metodu evaluateAll čija je deklaracija:

```
tPrData evaluateAll(string& gt_folder,string&
det_folder,vector<string> file_list,double min_overlap,string
class_name,DIFFICULTY dif);
```

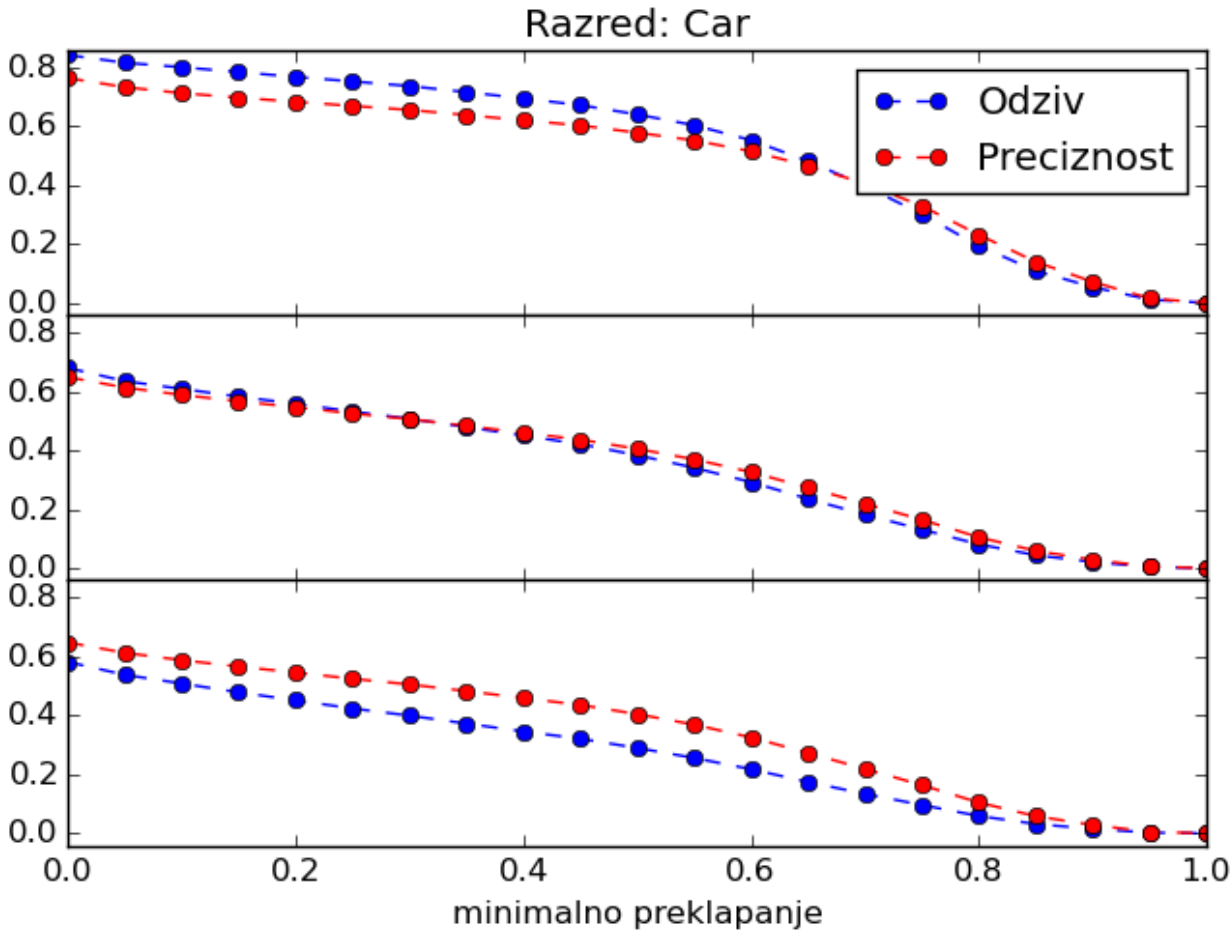
Ta metoda stvara novi objekt tipa tPrData koji čuva podatke o broju TP, FP i FN detekcija. Metoda za svako ime datoteke predano u listi datoteka `file_list` učitava datoteku s oznakama i njoj odgovarajuću datoteku s detekcijama, te za njih poziva metodu evaluate:

```
void evaluate(vector<tGroundtruth> gt,vector<tDetection>
det,double min_overlap,tPrData& stat,string
class_name,DIFFICULTY dif);
```

gdje je argument gt vektor koji čuva oznake pročitane iz datoteke oznaka,a det vektor koji čuva detekcije. Za svaku oznaku u vektoru gt traži se pripadna detekcija u vektoru det. To se provodi tako da se računa preklapanje sa svakom od detekcija i uzme ona s kojom je ostvareno najveće preklapanje, uzimajući u obzir da preklapanje mora biti veće od minimalno zadanog. Ukoliko detekcija nije *true positive* ali ima preklapanje s bilo kojom drugom oznakom neogovarajućeg razreda program za evaluaciju je ne računa kao *false positive* iz razloga što je ta dektekcija validna ali detektira objekt razreda koji se trenutno ne evaluira. Razlog takvom pristupu leži u činjenici da program za detekciju koji se u ovom radu evaluira ne razvrstava detekcije na razrede.
Program nakon što obradi sve datoteke na zaslon ispisuje broj true positive, false positive i false negative vrijednosti te izračunati odziv i preciznost

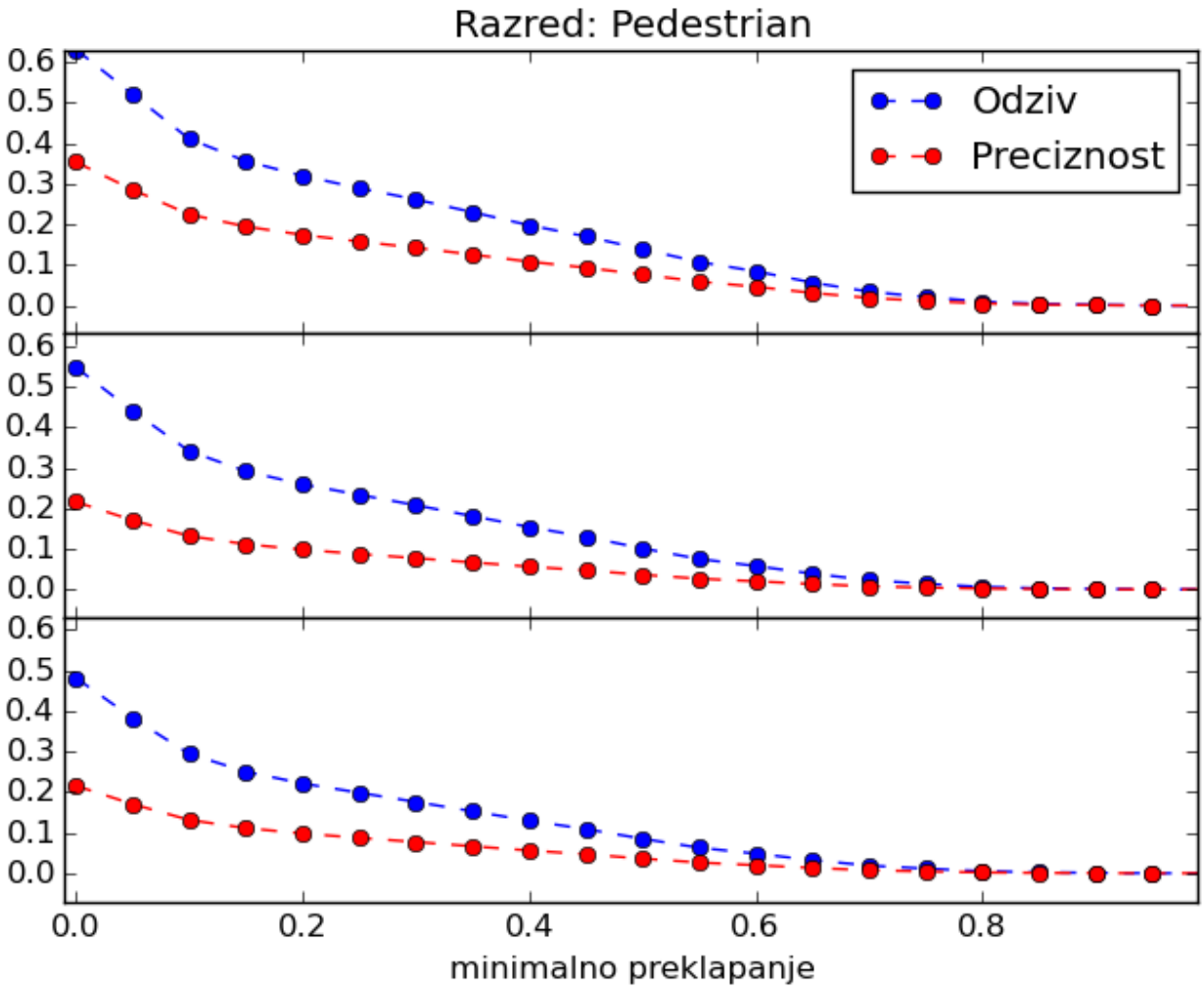
Evaluacija i rezultati

Prvi korak u evaluaciji bio je evaluacija nad cijelim skupom za različita minimalna preklapanja. Pri tome je interesno područje unutar oblaka točaka u rasponu: $y_{min} = 0.3m$, $y_{max} = 3m$, $x_{min} = -5m$, $x_{max} = 10m$, $z_{min} = 0m$ i Grafovi koji slijede prikazuju ovisnost preciznosti i odziva o minimalnom preklapanju za koje se detekcija prihvaća.



Slika 4.2. Graf ovisnosti odziva i preciznosti za razred Car

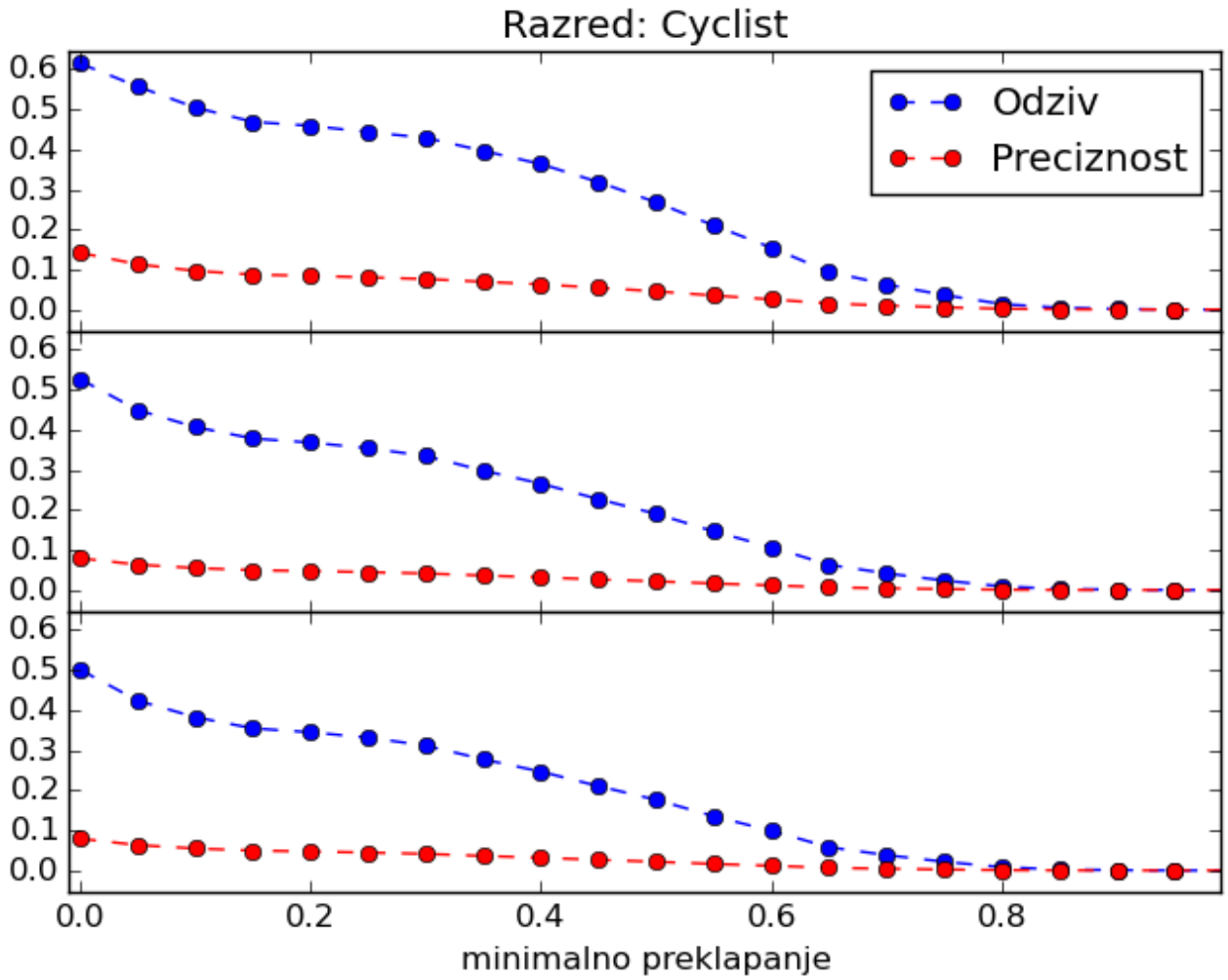
Slika 4.2 prikazuje tri grafa ovisnosti preciznosti i odziva o minimalnom preklapanju za razred Car s tim da je najviši podgraf prikaz rezultata za težinu evaluacije EASY, srednji za težinu MODERATE a donji za težinu HARD. Zbog toga je na najvišem grafu postignut najviši odziv.



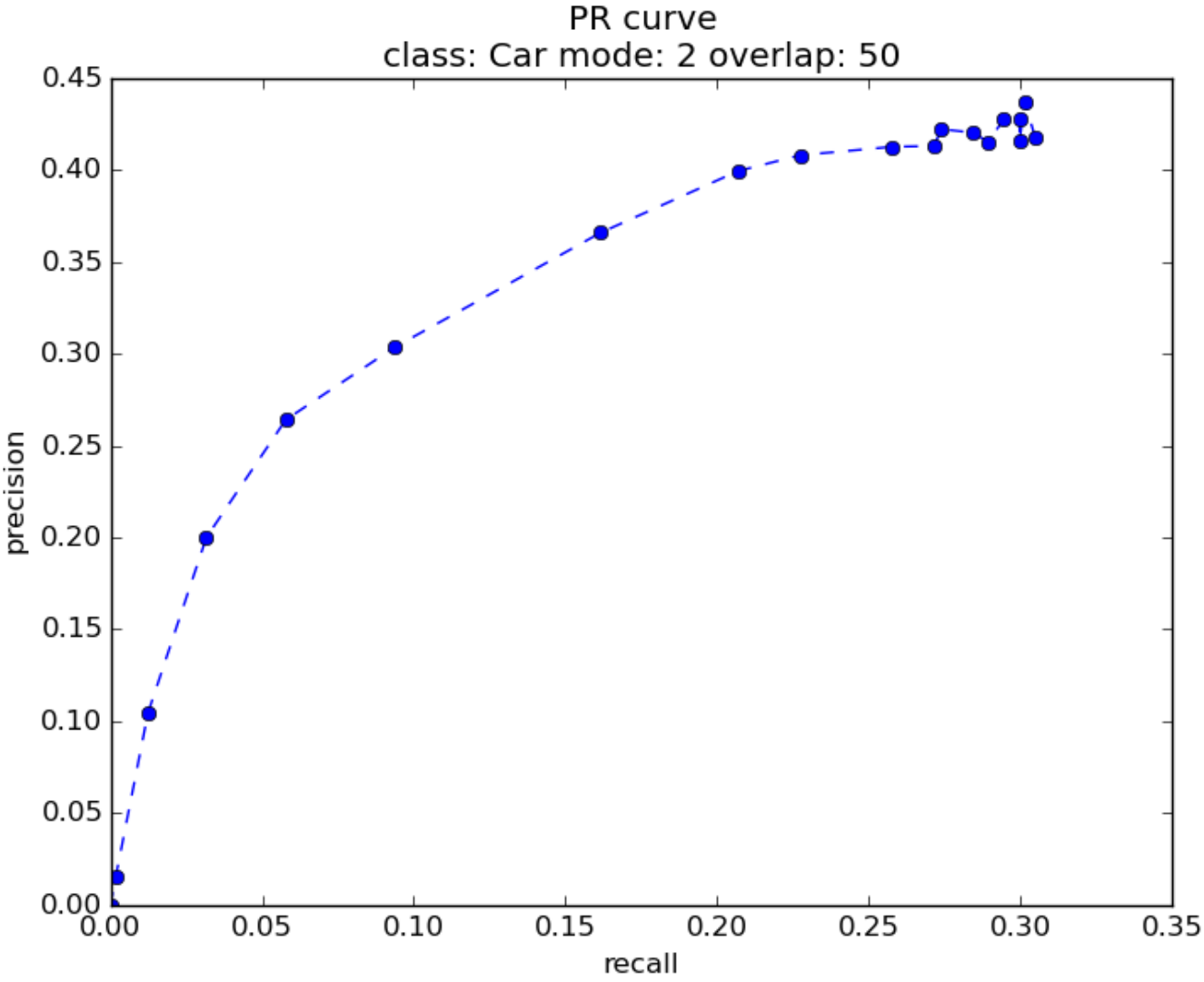
Slika 4.3. Graf ovisnosti odziva i preciznosti za razred Pedestrian

Slika 4.3. prikazuje ovisnost odziva i preciznosti za razed Pedestrian. U točki gdje je minimalno preklapanje 50% odziv i preciznost su relativno niski. Slika 4.4. prikazuje isti graf za razred Cyclist. Odziv je nešto lošiji u odnosu na razred Car ali bolji od odziva za razred Pedastrian. Odziv je općenito najviši za razred Car, naprosto je automobil ima najveće dimenzije.

Slika 4.4. Graf ovisnosti odziva i preciznosti za razred Cyclist



Drugi korak u evaluaciji bio je pronaći ovisnost preciznosti i odziva odnosno generirati *precision recall* krivulju. Da bi se to postiglo trebala sam pronaći parametar koji će kontrolirati broj detekcija. Kao taj parametar sam uzela raspon interesnog područja u oblaku točaka. Napisala sam skriptu koja varira te parametre te iterativno poziva program za detekciju te sprema tekstualne datoteke koje sadrže rezultate. Nakon toga skripta poziva program za evaluaciju i generira rezultate u obliku precision recall krivulja. Jedna takva krivulja prikazana je na slici. Ta krivulja rezultat je evaluacije za manji uzorak od 200 slika i 20 iteracija pri čemu se u svakoj iteraciji sužava interesno područje.



Slika 4.5. Graf međusobe ovisnosti preciznost i odzivai za razred Car

Validacija rezultata

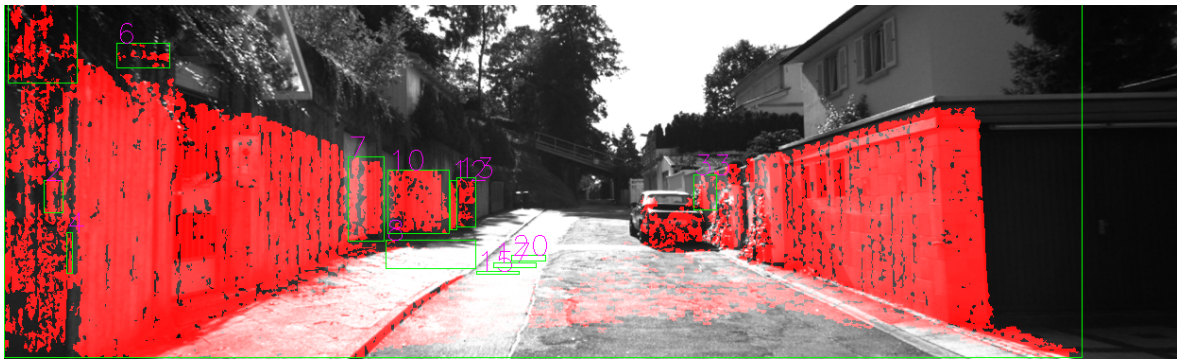
Analizirajući izlazne slike iz programa za detekciju uočila sam grupu slika na kojima nije označen niti jedan objekt pa čak ni šum. Izdvojila sam tridesetak takvih slika i pokušača variranjem raznih parametara dobiti bilo kakav odziv. Kako u slikama nije bio prisutan nikakav šum pokušala sam širiti raspon točaka koje se pretražuju u slikama. Odziv sam dobila za visoke vrijednosti po y i z osi iz čega sam zaključila da y os nije okomita na cestu odnosno da ravnina ceste nije dobro estimirana. To se događa jer metoda za estimaciju ravnine ceste, kao što je prije rečeno uzima u obzir samo određeni podskup točaka koje najbolje opisuju cestu.

Pod pretpostavkom da se automobil nalazi u desnom traku raspon koji daje zadovoljavajuće rezultate je 3m lijevo i 1.2m desno od kamere.

Slika 4.6. Parkirani automobili

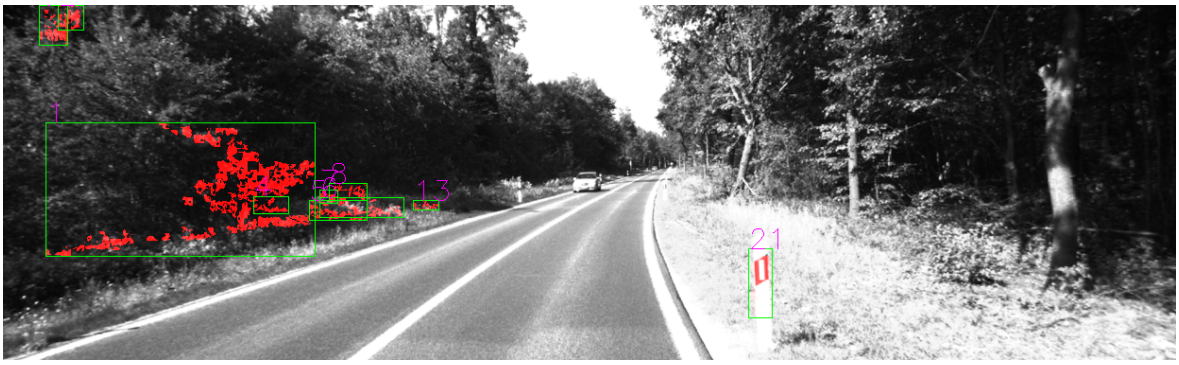
Slika 4.7. Kamion zaklanja lijevi dio ceste

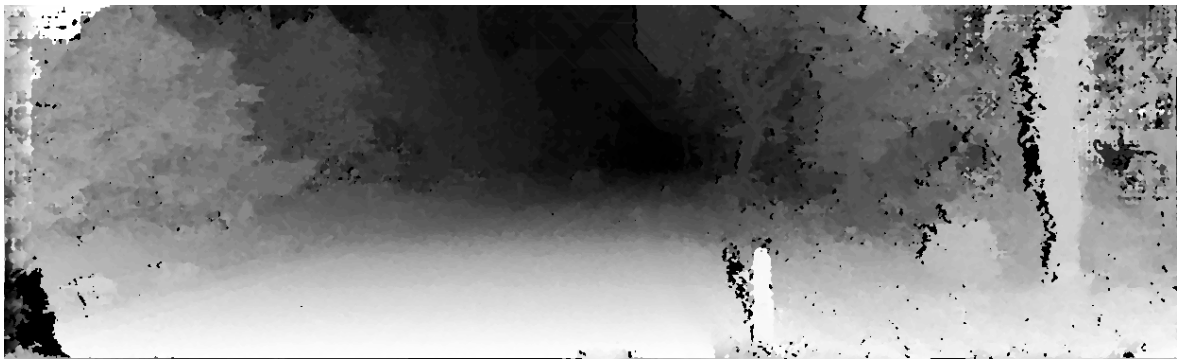
Ukoliko se automobil nalazi u potpuno drugačijim situacijama kao što je prikazano na slikama 4.5 i 4.6 uz tako uzete točke nije moguće procijeniti ravninu ceste odnosno detektirati objekte na slici. Pokušala sam jednostavno uzeti više točaka za procjenu ravnine ceste kako bih obuhvatila više slučajeva, ali to je negativno utjecalo na odzive u slikama u kojima je početna konfiguracija davala dobre rezultate. S obzirom da je u ispitnom skupu više slika u kojima se automobil nalazi u desnoj traci podesila sam metodu za procjenu ravnine ceste da prima parametre koji odgovaraju toj situaciji. Osim tog problema, problem je i već spomenuta situacija u kojoj se cesta ili zidovi detektiraju kao jedan veliki objekt prikazana na slici 4.7.



Slika 4.8. Zidovi i cesta detektirani kao jedan objekt

Osim ta dva problema postoje situacije i u kojima nije došlo do detekcije jer se objekt stapa s pozadinom i na samoj mapi dispariteta ga je gotovo nemoguće razlikovati od šuma kao što je prikazano na slici 4.9.





5. Zaključak

Cilj ovog rada bio je upoznati se s programom kojeg je izgrađen u magistarskom radu (Braut, 2014.) i podesiti ga da radi s metodum poluglobalnog podudaranja. Sljedeći korak bio je upoznati se s KITTI object ispitnim skupom te na njemu evaluirati rad programa. U tu svrhu izgrađen je jednostavan program za evaluaciju. Program kao ulaz prima skup oznaka i skup detekcija koje uspoređuje i kao rezultat vraća izračunati odziv i preciznost. Uz to napisane su skripte koje omogućuju automatizaciju pokretanja programa za detekciju i evaluaciju te generiranje rezultata.

Rezultati evaluacije prikazani su u obliku grafova koji prikazuju ovisnost preciznosti i odziva o minimalnom preklapanju te pokazali su da je odziv i preciznost programa relativno niski čak i pri malim preklapanjima. Uočeno je da je detekcija osjetljiva na estimaciju ravnine ceste pri čemu loša estimacija može u pojedinim slikama dati odziv jednak nuli ili detektirati cestu kao jedan veliki objekt. Iako metoda poluglobalnog podudaranja daje kvalitetniju mapu dispariteta, postoje situacije u kojima unatoč tome nije moguće detektirati objekt. U vidu poboljšanja moglo bi se pokušati kao što predlaže rad (Braut, 2014.) korištenjem usmjerenosti normala točaka, odvojiti objekte različito usmjerenih normala.

Možda bi se program mogao proširiti tako da se omogući da program svoje detekcije razvrsta u razrede i svakoj detekciji dodatno pridruži broj [0,1.0] koji predstavlja u kojoj je mjeri siguran u detekciju. Rezultate tog programa bilo bi zanimljivo poslati na KITTI evaluacijski server.

LITERATURA

Viktor Braut. Detekcija objekata u gustoj stereoskopskoj rekonstrukciji. Magistarski rad, Fakultet elektrotehnike i računarstva, Zagreb, Hrvatska, 2014.

Slavko Grahovac. Pronalaženje prohodnog tla stereoskopskim računalnim vidom. Magistarski rad, Fakultet elektrotehnike i računarstva, Zagreb, Hrvatska, 2015.

Ivan Krešo. Napredno estimiranje strukture i gibanja kalibriranim parom kamera. Magistarski rad, Fakultet elektrotehnike i računarstva, Zagreb, Hrvatska, 2013.

Daniel Scharstein, Richard Szeliski. A Taxonomy and Evaluation of Dense Two Frame Stereo Correspondence Algorithms, http://vision.middlebury.edu/stereo/taxonomy-IJCV.pdf

Lokalizacija objekata primjenom stereoskopske rekonstrukcije

Sažetak

Rad razmatra postupak lokalizacije objekata iz oblaka točaka dobivenog stereoskopskom rekonstrukcijom. Rad se nastavlja na diplomski rad Viktora Brauta koji je razvio računalni program podešen za detekciju objekata iz para rektificiranih slika snimljenih stereo sustavom montiranim na krov vozila. U okviru rada korištena je metoda poluglobalnog podudaranja radi dobivanja gušće disparitetne mape u odnosu na lokalne metode. Rad programa je evaluiran na KITTI object ispitnom skupu. Rezultati su prikazani u obliku krivulja koje pokazuju ovisnost odziva i preciznosti o traženom minimalnom preklapanju i u obliku krivulja koje prikazuju međusobnu ovisnost preciznosti i odziva.

Ključne riječi: računalni vid, stereoskopska rekonstrukcija, oblak točaka, lokalizacija, detekcija, evaluacija, preklapanje, odziv, preciznost

Object localization in dense stereoscopic reconstruction

Abstract

The thesis considers the method of object localization in a point cloud. This thesis uses software for object localization developed by Viktor Braut and continues experiments described in his master thesis. This software is specifically tuned for the object detection in the rectified image sequences recorded by a pair of calibrated cameras mounted on the vehicle. The stereo matching method used for disparity map computation is Semi global block matching because it gives a better estimation of scene than local methods. Detection program was evaluated on KITTI object detection benchmark and the graphs presenting results are shown. First type of graph shows recall and precision values as a function of minimal overlapping required and the second type of graph shows precision recall curve.

Key words: computer vision, stereoscopic reconstruction, point cloud, localization, detection, evaluation, overlapping, recall, precision